

This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

**As rescanning documents *will not* correct images,
please do not report the images to the
Image Problem Mailbox.**

Requested Patent: JP2000029731A

Title:

A DUAL-MODE VLIW ARCHITECTURE WITH SOFTWARE-CONTROLLED
PARALLELISM ;

Abstracted Patent: EP0962856, A3 ;

Publication Date: 1999-12-08 ;

Inventor(s): SHIELL JONATHAN H (US); BARTLEY DAVID H (US) ;

Applicant(s): TEXAS INSTRUMENTS INC (US) ;

Application Number: EP19990201702 19990528 ;

Priority Number(s): US19980088314P 19980605 ;

IPC Classification: G06F9/38 ;

Equivalents: ;

ABSTRACT:

This invention is a very long instruction word data processor (200) including plural data registers (140A, 140B), plural functional units (130A, 130B) and plural program counters (110A, 110B) and is selectively operable in either a first or second mode. In the first mode, the data processor executes a single instruction stream. In the second mode, the data processor executes two independent program instruction streams simultaneously. In the second mode the data processor may respond to two instruction streams accessing only corresponding halves of the data registers and function units. Alternatively, the data processor may respond to a first instruction stream including instructions referencing the whole data processor employing A side function units by alternatively dispatching (1) instructions referencing the A side data registers and the A side function units and (2) instructions referencing the B side data registers and the B side function units. In the first mode, the data processor fetches N bits of instructions each cycle. In the second mode the data processor may fetch N bits of instructions for alternate program counters on alternate cycles or fetches N/2 bits of each of the first and second program counters. The data processor includes interrupt steering and masking control logic (125) allowing instructions to control whether the first instruction stream or the second instruction stream receives interrupts.

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2000-29731

(P2000-29731A)

(43) 公開日 平成12年1月28日 (2000.1.28)

| (51) Int.Cl. ⁷ | 識別記号 | F I | ターコード (参考) |
|---------------------------|-------|--------------|------------|
| G 0 6 F 9/46 | 3 6 0 | G 0 6 F 9/46 | 3 6 0 B |
| 9/30 | 3 5 0 | 9/30 | 3 5 0 F |
| 9/38 | 3 7 0 | 9/38 | 3 7 0 B |
| | | | 3 7 0 X |

審査請求 未請求 請求項の数 1 OL (全 12 頁)

(21) 出願番号 特願平11-158532

(22) 出願日 平成11年6月4日 (1999.6.4)

(31) 優先権主張番号 0 8 8 3 1 4

(32) 優先日 平成10年6月5日 (1998.6.5)

(33) 優先権主張国 米国 (US)

(71) 出願人 590000879

テキサス インストルメンツ インコーポ
レイテッド

アメリカ合衆国テキサス州ダラス、ノース
セントラルエクスプレスウェイ 13500

(72) 発明者 パートレイ エイチ. デビッド

アメリカ合衆国 テキサス、ダラス、エコ
ー リッジ コート 10235

(72) 発明者 シェル エイチ. ジョナサン

アメリカ合衆国 テキサス、プラノ、ロン
グフェロウ ドライブ 4300

(74) 代理人 100066692

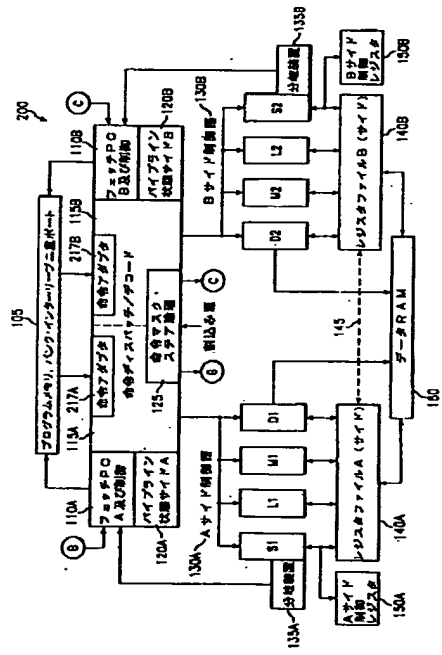
弁理士 浅村 皓 (外3名)

(54) 【発明の名称】 命令レベル及びタスクレベル並行処理のソフトウェア制御可変ミックスを与える2重モードVL
1Wアーキテクチャ

(57) 【要約】

【課題】 命令レベル及びタスクレベルの並行処理の2
重モードをソフトウェア制御により切り替え可能な超長
命令語アーキテクチャのデータプロセッサを提供する。

【解決手段】 超長命令語データプロセッサは、複数デ
ータレジスタと、複数機能装置と、複数プログラムカウ
ンタとを含み、第1又は第2モードのどちらかで選択的
に動作可能である。第1モードでは、データプロセッサ
は単一命令ストリームを実行し、第2モードでは、デー
タプロセッサは2つの独立したプログラム命令ストリー
ムを同時に実行する。第2モードでは、データプロセッ
サはデータレジスタと機能装置の対応する半分のみア
クセスする2つの命令ストリームにตอบสนองしてもよい。



【特許請求の範囲】

【請求項1】超長命令語データプロセッサにおいて、
 複数データレジスタと、
 複数機能装置と、
 複数プログラムカウンタと、
 を含み、前記データプロセッサは、
 プログラムカウンタの所定のものの制御下で、データレジスタの選択されたものと機能装置の選択されたものによりプログラム命令が実行される第1モードと、
 第1群のデータレジスタと第1群の機能装置を使用して第1プログラムカウンタがプログラム命令の実行を制御し、前記第1群のデータレジスタから分離した第2群のデータレジスタと前記第1群の機能装置から分離した第2群の機能装置を使用して第2プログラムカウンタがプログラム命令の実行を制御する第2モードと、に従って選択的に動作可能な、超長命令語データプロセッサ。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明の技術分野はマイクロプロセッサとデジタル信号プロセッサ、特に超長命令語(VLIW)を使用するものである。

【0002】

【従来の技術】本発明はVLIW(超長命令語、Very Long Instruction Word)マイクロプロセッサ及びデジタル信号プロセッサ(DSP)の分野に関する。この型式のプロセッサは、現在のアプリケーションで利用可能である場合には、高度の命令レベル並行処理(instruction level parallelism, ILP)を利用する能力により特徴付けられる。これらのプロセッサ用に開発されたソフトウェア・アプリケーションは変化する度合いの潜在ILPを示す。命令レベル並行処理とは、命令が互いに独立で、直列にはなく並行に実行可能である程度である。潜在ILPの低いアプリケーションは常にVLIWプロセッサの全能力を活用できない。タスクレベル並行処理(TLP)が代わりに利用可能である場合には、低い潜在ILPを有する実行期間にたいする性能が改善できることもしばしばある。タスクレベル並行処理とは、1つ以上の独立したタスク又は命令スレッドを同時に実行するプロセッサの能力を指す。このような独立したタスクは標準的には独立した問題に向けられる少なくとも数十又は数百の命令を含む。VLIWプロセッサの大きな利用法は、低潜在ILP時にはプロセッサの多くの機能装置はアイドルであり、従って他のタスク(又はスレッド)からのコードを実行するのに使用可能であることから来している。

【0003】

【発明が解決しようとする課題】VLIWはデジタル信号処理の性能に敏感な面や、その他の「数値計算専用(number crunching)」アプリケーションに標準的なもののような通常の、ループ指向のタスクに特に有効である。多くの最近のアプリケーションは、1個のプロセッサがプ

ログラム・パラダイムの混合を処理できることを必要としている。例えば、実時間埋め込み型DSPアプリケーションはDSPと制御処理タスクの両方をミックスする。後者のタスクは通常殆ど潜在ILPを有していない。複数スレッド実行は、時間が重要なDSPカーネル内部ループ実行にのみ関係していない時、アプリケーションにより適合している。

【0004】この問題は多数の方法で取り組まれた。従来技術の1つの例は、ケックラー他による「並行処理用の統合コンパイルと実行持スケジュールによる複数プロセッサ結合システム」という名称の米国特許第5,574,939号に示すような複数スレッドへのVLIW方式である。ケックラー他は、コンパイル時に単一VLIW語に相互ミックスされた複数スレッドを実行可能なVLIWシステムを示している。この方式では、別個のプログラム・カウンタを必要とするであろう多数の異なる命令ストリームが共に静的にスケジュールされ、単一のプロセッサの制御下で単一の組合せ命令ストリームとして実行される。

【0005】スーパースケーラ・プロセッサ(これはハードウェア組み立てのVLIWシステムの1形式と考えることもできる)に対しては、複数の時間インターリーブされたコード・ストリームの実行が提案されている。最初に、ストリームAからN命令がフェッチされ、次いでBからN命令が、又は処理がストール(stall)するまでAからフェッチし次いで処理がストールするまでBに切替える、等である。この型式のプロセッサ動作の例は、セルトルダ他による「2つの独立した命令ストリームからの命令を直列化する方法と装置」という名称の米国特許第3,771,138号に示されている。セルトルダ他は時間多重方式で実行される2重命令バッファによるプロセッサを教示している。ロバート他による「2重シーケンサ・マイクロプロセッサ」という名称の米国特許第4,320,453号も時間多重プロセッサによる複数スレッドを教示している。デネロコーからの1次VLIW特許、ジラード他による「同時タスク及び命令プロセッサと方法」という名称のHEP特許米国特許第4,222,790号も時間多重化を基にしている。

【0006】加えて、VLIWプロセッサは特に割込みに適していない。割込みハンドラは殆どILPを示さず、プロセッサ状態を割込みに切替えると、VLIWの実行は高価になり遅い。さらに、割込みルーチンは非常に短時間のみ通常実行する。従って、VLIWプロセッサは割込み処理時にはリソースを標準的には浪費する。

【0007】

【課題を解決するための手段】本発明は、複数データレジスタ、複数機能装置及び複数プログラムカウンタを含む超長命令語データ・プロセッサである。データ・プロセッサは第1又は第2モードのどちらから選択的に動作する。第1モードでは、プログラム命令は所定のプログ

ラムカウンタの制御下でデータレジスタの選択したものと機能装置の選択したものにより実行される。第2モードでは、第1プログラムカウンタは第1群のデータレジスタと第1群の機能装置を使用してプログラム命令の実行を制御し、第2プログラムカウンタは分離した第2群のデータレジスタと分離した第2群の機能装置とを使用してプログラム命令の実行を制御する。

【0008】複数のデータレジスタはAサイド半分とBサイドの分離した半分から構成され、複数の機能装置はAサイドの半分とBサイドの分離した半分から構成されることが望ましい。第1プログラムカウンタはAサイドのデータレジスタとAサイドの機能装置に関係し、第2プログラムカウンタはBサイドのデータレジスタとBサイドの機能装置とに関係する。第2モードではデータプロセッサは2個の独立したプログラム命令ストリームを同時に処理可能である。データプロセッサは複数の制御レジスタ、少なくとも1個の制御レジスタがAサイドとBサイドに重複している、を含む。第1モードでは、動作は重複制御レジスタのAサイド部分により制御される。第2モードでは、重複制御レジスタのAサイド部分がAサイド動作を制御し、重複制御レジスタのBサイド部分がBサイド動作を制御する。

【0009】データプロセッサは命令又は制御レジスタの状態を介して第1モードから第2モードへ、そして第2モードから第1モードへ変更できる。

【0010】第2モードでは、データプロセッサは、各々がデータレジスタと機能装置の対応する半分のみにアクセスする2つの命令ストリームにตอบสนองする。または、(1) AサイドのデータレジスタとAサイドの機能装置を参照する命令と(2) BサイドのデータレジスタとBサイドの機能装置を参照する命令とを交互にディスパッチすることにより、データプロセッサはAサイド機能装置を使用して実行される任意のデータレジスタと任意の機能装置を参照する命令を含む第1命令ストリームにตอบสนองする。データプロセッサは、第1命令ストリームにより使用されるBサイドのデータレジスタの数と数が等しいB'サイドのデータレジスタの組を含んでもよい。この別案では、データプロセッサは、Bサイドの機能装置を使用して実行されるデータプロセッサの半分と機能装置の半分を参照する命令のみを含む第2命令ストリームにตอบสนองする。

【0011】データプロセッサは命令を記憶するプログラムメモリを含む。第1モードでは、データプロセッサは第1プログラムカウンタに対応する各サイクルでプログラムメモリからNビットの命令をフェッチする。第2モードでは、データプロセッサは交互のサイクルで交互のプログラムカウンタにNビットの命令をフェッチする。又は、第2モードでは、データプロセッサは第1プログラムカウンタに対応するN/2ビットの命令をフェッチし、かつ第2プログラムカウンタに対応するN/2ビッ

トの命令をフェッチする。

【0012】データプロセッサは、第1命令ストリーム又は第2命令ストリームのどちらかが割込みを受入れるかを制御する命令を可能とする割込みステア及びマスク制御論理を含む。

【0013】

【発明の実施の形態】本特許に図示する望ましい実施例は、以後C6xプロセッサ又はC6xと呼ぶテキサス・インストラメント社TMS320C62x/C67xファミリのプロセッサを基にし、その基本知識を想定している。さらなる情報は、共にテキサス・インストラメント社から市販されている「TMS320C62x/C67x CPUと命令セット、参照ガイド」と「TMS320C62x/C67x技術概略」を参照されたい、この両者は引用により含まれる。説明の都合上、以下ではC6x仕様の一部であるいくつかの特徴と拡張は削除される。これら又はその他の拡張は当業者には容易に明らかとなる。

【0014】ここで図1に戻ると、C6xの例示中央処理装置(CPU)100のコアとプログラムメモリ105とデータメモリ160の簡略化した図が図示されている。ここで当該技術において公知の例示VLIWプロセッサの動作を復習する。C6xは対称VLIWプロセッサ、すなわちAサイド130AとBサイド130Bの2サイドがあることに注意されたい。Aサイド130AとBサイド130Bの両方が、S、L、M及びD装置と呼ばれる4個の機能装置と各レジスタファイル140Aと140Bを含む。一般にC6xでは、S装置はシフト演算を実行し、L装置は論理演算を実行し、M装置は乗算を実行し、D装置はデータのロードとストアを実行する。加えて、この型式のプロセッサは一緒にまとめた複数命令を使用する。ソフトウェア制御(通常コンパイラ)下で、命令は単一VLIWとして8個のブロックにまとめられる。これら8命令のブロックがVLIWパイプラインの複数機能装置にディスパッチされる。C6x用語での単一のVLIWは命令パケットと呼ばれる。

【0015】命令ディスパッチ/デコード装置115のフェッチ・プログラムカウンタ(PC)及び制御装置110によりアドレスされる命令パケットは、プログラムメモリ105から命令ディスパッチ/デコード装置115にフェッチされる。命令ディスパッチ/デコード装置115はこれらの部分命令をデコードし、スケジュールし、最後に命令により指示されるように特定の機能装置(S、L、M又はD)とサイドへディスパッチする。8命令までがサイクル当たりディスパッチされ、サイド当たり4、機能装置あたり1である。サイド130Aと130Bはプログラム分岐を処理する各分岐装置135Aと135Bを有する。分岐が生じる場合、対応する分岐装置135A又は135Bはフェッチ・プログラムカウンタ及び制御装置110に指示して新たなアドレスからのフェッチを開始させる。割込み処理は割込みマスク及びステア装置125で発生する。割込みが生じた場合、割込みマスク

及びステア装置125はフェッチ・プログラムカウンタ及び制御装置110に指示して対応する割込みハンドラのアドレスからフェッチを開始させる。複数の分岐及び/又は割込みが1サイクルで生じた場合、優先方式を使用して1つを選択し、他の事象は無視されるが又は遅延される。パイプライン状態装置120は、プログラムカウンタやパイプの各段の状態情報のようなパイプライン・プロセッサで保持される標準的な状態情報から構成される。

【0016】実行の前に、Aサイド130A又はBサイド130B上の各命令は関係するレジスタファイル140A又は140Bからそのレジスタ引数を読取らせる。加えて、レジスタ交差路145を介してサイドあたり1つの交差レジスタ読取りが可能である。各サイドの実行はデータメモリ160を共有し、両サイド130Aと130Bはデータメモリ160を独立にアクセスする。制御レジスタ150はC6xプロセッサ100の動作を制御する際に使用するためのその他の情報を提供する。

【0017】従って、このシステムは、従来の技術で示すように、単一の混合されたストリームで与えられるのでなければ、1時に1命令ストリームのみを処理可能である。図2は本発明の1実施例を図示する。別々のフェッチ・プログラムカウンタ及び制御装置110Aと関係する制御部が各Aサイド130AとBサイド130Bに設けられる。プログラムメモリ105から命令ディスパッチ/デコード装置115に2つのポートもある。命令ディスパッチ/デコード装置115は2つの部分115Aと115Bに分離された。各装置115Aと115Bには命令アダプタ217Aと217Bが追加された。命令アダプタ217Aと217Bは各々Aサイド130AとBサイド130Bに命令を供給する。又、分岐装置135Bはフェッチ・プログラムカウンタ及び制御110(図1)の代わりにフェッチ・プログラムカウンタ及び制御装置110Bに値を与える。加えて、フェッチ・プログラムカウンタ及び制御装置110Aと110Bの下全ての他のパイプライン状態も、装置120Aと120Bとして、AサイドとBサイドの両方に分離(すなわち、重複)している。制御装置150Aと150Bも同様である。

【0018】C6xの対称2進分離を示しているが、プロセッサは2つ以上の非対称部分(ある部分は互いに対称であるが他とは異なる)にも分離可能である。C6xにはこれが最良の分離であり、何故ならレジスタファイルがサイクル時間と相互接続のコストの理由から既にA及びBサイドに分離され、各サイドの機能装置は相当に対称だからである。本発明の望ましい実施例は、現在のC6xにはない各種の機能装置の命令に対称性を与えることに注意されたい(15ビット・オフセット付きロード/ストア又は読取り/書き込み制御レジスタのような)。絶対に必要な条件ではないが、これはサイド独立性によりコード化が容易になり、コードを一方のサイドにのみ固定す

る必要条件を防止できる。

【0019】図2に図示するプロセッサは以下のように動作する。フェッチ・プログラムカウンタ及び制御装置110Aと110Bの各々が、プログラムメモリ105(これは2重ポート化されている)から各命令ディスパッチ/デコード装置115Aと115Bへ命令の半パケット(VLIW8命令パケットの1/2)をフェッチする。命令アダプタ装置217Aと217Bは命令をA及びBサイドへ各々バインドする。次いで各サイドのディスパッチ及びデコード論理115Aと115Bは「通常の」C6xにより実行用の半パケットを用意する。その後命令が各サイドにディスパッチされる(Aサイドの命令はA機能装置S1、L1、M1、及びD1へ、B命令はB装置のS2、L2、M2及びD2へ)。割込みや分岐のような制御流れの変更の処理を除いて、ディスパッチからプロセッサは「通常の」C6xとして動作する。

【0020】分離プロセッサでは、分離していない割込みマスク及びステア装置125はAサイド130A制御レジスタとBサイド130B制御レジスタの両方から割込みマスクを受取る。割込みマスク及びステア装置125はどちらのサイドが割込みを付勢されているかを検査し、これに従って割込みを与える。Aサイド130AとBサイド130Bの両方が割込みを付勢されている場合、割込みマスク及びステア装置125はこれらの割込みを優先度付けして、一方のサイドがその割込みを受取り他方のサイドでは割込みは停止される。他の実施例では一方のサイドのみが割込みを処理する。この場合、そのサイドの割込みマスクのみが検査される。この実施例はその簡潔性のため多くの場合に望ましい。両方のサイドが割込みを処理するよう動作可能な場合、どちらのサイドがどちらの割込みを見ることができるかを制御するため、割込みステア論理及び関連マスクが「マスタ」サイド(タスク)を可能とすることもまた望ましい。このステア論理は割込みの初期マスクとしても動作する、すなわちあるサイドが割込みを見るためには、そのサイドの割込み付勢レジスタの対応する構成のマスクとその割込みのステア・マスクの両方がこれを可能としなければならない。

【0021】2つのサイドに全ての通常C6x制御レジスタを複製する必要はない。多くの実施例で、制御レジスタのあるもののみが2重コピーを有し、その他の制御レジスタは「マスタ」サイドにのみ存在する。「スレーブ」サイドがコピーを有していない制御レジスタに書き込み(又は何らかのシステム読取り)しようとする場合、割込みを与えることが通常望ましい。C6x実施例で2重化される制御レジスタ又は制御レジスタの一部は以下を含む: アドレッシング・モードを制御するAMR; 飽和及びエンディアン制御ビットを含むCSRで、その他のビットはスレーブ・サイドでは非2重化読取り専用に出来る; 実行命令パケットのアドレスを記憶するPE1、である。

「マスタ」及び「スレーブ」サイド・モデルを使用する場合、C62xで割込みを処理するその他の制御レジスタは2重化される必要はない。C67x(浮動小数点版)では、浮動小数点制御を処理する別の制御レジスタも2重化されなければならない。

【0022】ここで命令アダプタ217Aと217Bの動作に戻ると、分離モードで使用する意図の命令はAサイド命令として書きこまれるのが望ましい。従って、Aサイド130A対Bサイド130Bを選択する全ての欄は交差ファイル読取りを除いてAサイド130Aを選択するよう設定される。実行時に、Bサイド命令アダプタ217Bはこれらの欄をBサイド選択に変換するが、Aサイド命令アダプタ217AはBサイド選択をAサイド選択に変換する。多くの場合これに必要な全ては簡単なビット反転(すなわち、「0」を「1」に、又は「1」を「0」に)であることに注意されたい。これは必要に応じてOR関数又はNAND関数により実行可能である。

【0023】新たな命令又は1個以上の制御レジスタ150A又は150Bのロードのような、本発明の分離モードを起動する各種の方法が可能である。後者が望ましい方法で、以下は「分離」制御レジスタの内容の例である：

ビット 欄

31： 分離半ストリームAと半ストリームBを付勢(上述した第1の実施例)。

30： 交差ファイル述語と読取りを付勢(ビット31セット時のみ有効)

29： 交差ファイル書込みを付勢(ビット31セット時のみ有効)

28： 全ストリームAと半ストリームBを付勢(後述する第2実施例)

27： 割込みステア機能を付勢(0の場合全ての割込みはBへ)

26-16： 予約

15-0： 割込みステア・マスク・ビット、IER制御レジスタにより割込み当り1ビット(「0」はAサイド130Aへ与えられ、「1」はBサイド130Bへ与えられる)

【0024】割込みステア・マスクはここでは1ビット・マスクとして与えたが、別案として割込み当り2ビットのマスクも使用可能であり、その場合第1ビットがAサイドへの提示を制御し、第2ビットがBサイドへの提示を制御する。

【0025】以下は、本発明により変更されたC6xシステムが、図1に示すような単一のPCモードで単一のスレッドを実行する状態から図2にあるような2重PCモードの2重スレッドを実行している状態へ如何に変換するかの例である。プログラムはちょうど内部ループを終了し、これからある種の制御型式コードを実行し割込みを検査するものと仮定する。

【0026】最初にプログラムは、プロセッサが分離されることを指示する値を制御レジスタ「分離」に書込む。同一サイクルで新規A及びBサイド・コードへプログラム実行がジャンプするものが発行され、これは新たな分離モードプログラムカウンタ値を設定する。次いで「分離」の発行と分離が生じる時との間のサイクルで、プログラムは通常モードの制御レジスタからAMRとCSR制御レジスタの内容をAサイド制御レジスタ150A又はBサイド制御レジスタ150Bのどちらかへコピーする。パイプライン・アドレスレジスタPE1はハードウェア・パイプラインによりセットされることに注意されたい。その他の必要なレジスタもこの期間にコピーされる。

【0027】分離したAサイド130AとBサイド130B命令ストリームが実行する最初の演算は、起動プログラムが既に実行しているか又は内容を将来必要としないかでない限り、その各々のレジスタファイルの内容の保存する。これの1つの例外は、新たなコード・ストリームが制御レジスタ内容(特にAMR)を変更する必要がある場合で、これが最初に実行されるか、又は起動プログラムによる分離前設定の一部として実行される。汎用レジスタを保存した後、「マスタ」サイドは割込みをホールドするか又は選択した割込みマスクを付勢する。割込みが発生した場合、「マスタ」サイドのみが影響され、他のサイドはその処理を不変のまま続行する。

【0028】以下は上記流れの連続した例で、ストアダブル語(レジスタnとn+1)を想定している。より高速の切替えが必要な場合、ストアダブル語命令はストア4倍語命令(レジスタn、n+1、n+2及びn+3)に拡張可能である。

1. 必要に応じて割込みを減勢、分離AとBサイドのコード・ストリームのアドレスを設定、新たな分離制御/割込みステア制御値を設定、非分離コード・ストリームの再開アドレスをメモリにストア(保存)。
2. 「分離」制御レジスタにその新たな分離制御/割込みステア制御値をロード(この演算と共に以下の2つのジャンプは「終了」に5サイクルかかる)、AサイドにAサイド分離コード・ストリームへのジャンプを発行、BサイドにBサイド分離コード・ストリームへのジャンプを発行、R0a(Aサイド汎用レジスタ0)とR1aをメモリのold_tsk_GRsにストアダブル(保存)、R0bとR1bをストアダブル、
3. AMR制御レジスタの内容をR0bへ移動、R0aとR1aの新たな値のロードを発行(終了に4サイクルかかる)、R0bとR1bの新たな値のロードを発行。
4. CSR制御レジスタの内容をR1bへ移動、R0bの内容をAサイドのAMRへ移動、R2aとR3aをストアダブル、R2bとR3bをストアダブル。
5. IER制御レジスタの内容をR0bへ移動、R0bの内容をAサイドのCSRへ移動、R6bをR2aへ移動、R0bとR1bをメモリのCtl_reg_saveへストアダブル、R4aとR5aをストアダ

ブル。

6. ISTP制御レジスタの内容をR1bへ移動、R7bをR3aへ移動、R6aとR7aをストアダブル、R4bとR5bをストアダブル。

7. R0bとR1bをメモリの(Ctl_reg_save+8)へストアダブル、R2aとR3aをストアダブル。(実際にはR6bとR7b)

8. ここでマシンは分離モードで実行中。

【0029】2つのサイドが単一のスレッドに再結合される、逆の「結合」操作も、プロセッサが再結合されるべきであることを指示する値を書込むことによりこの場合の「分離」制御レジスタを変更することで優先的に実行される。再び望ましい実施例では、制御レジスタのロードと共に一方のサイドからジャンプが発行される、通常の使用は制御レジスタを書込むサイドもジャンプを発行することに注意されたい。その他の実施例はジャンプなしで「結合」を発行し、「結合」が生じた時に制御レジスタに書込んだサイドからフェッチ・プログラムカウンタ及び制御装置110A又は110Bが引き継ぐか、又は「結合」後に固定サイド(例えばマスタ・サイド)が常に引き継ぐか、である。制御レジスタに書込むことにより作動される「結合」の代わりに場合には、1つ以上の特殊化命令、又は当該技術において既知のその他の方法で実行可能である。

【0030】望ましい実施例ではプログラムは汎用レジスタと制御レジスタの保存とロードに責任を負う。そこで結合時に所要値が分離時のマスタ・コピー内容と異なる場合、プログラムは所要値をロードする必要がある。分離モードで実行していない時には、プロセッサは制御レジスタのマスタ・コピーのみを使用することに注意されたい。また、分離モードを出た時には、割込みステア機能は意味がないため無視される。その他の提供された分離モードのみの制御機能に対してもこれは成立する。

【0031】2体分離以上が提供された場合、例えば、2体又は4体分離が可能であるが、各々の個別の制御を提供することが望ましいことに注意されたい。説明の都合上、以後4体分離の4部分をA、B、C、Dと呼ぶ。2体分離の組合せ部分はABとCD、ABCとD又はAとBCDである。従ってプロセッサを2又は3(AB/C/D、A/BC/D又はA/B/CD)又は4部分に分離できる。対応して、4体の結合は、4個の分離した装置から未分離の2体又は3体形式に戻りうる。

【0032】分離の1つの特殊な機能はC6xアーキテクチャにより可能な交差ファイル読取り、書込み及び述語(predication)である。本望ましい実施例では、これらは分離により変化せず動作し、プロセッサの2サイドの命令ストリーム間で非常に高速な形式のプロセス間通信を可能とする。この例は、Aサイドがそのレジスタの1つがゼロにされているかをボールし(例えばジャンプのような述語(predicated)演算)、Bサイドは所定

の条件が発生した時にレジスタをゼロにする、ような場合である。これはもちろん、2つのタスクの命令ストリームが非常に緊密に結合し、常に共に実行するか又はコードのこの部分が実行している時には少なくとも共に実行していることを想定している。これがそうでない場合、以下のような問題が発生する：Bがループカウンタとしてゼロにしたレジスタを使用して異なるコード・ストリームがAサイドで実行していて、次いでBサイドがレジスタをクリアしたためループを以前に出してしまった、又は異なるサイドから同じレジスタへの2つの書込みが生じる場合があり、これはハードウェアの問題を生じる場合がある。他の代案は、交差ファイル操作、又は少なくとも交差ファイル書込みを禁止し、発生した場合に例外(割込み)を処理することもできる。

【0033】第1実施例の1つの欠点は、異なるストリームとして分離ストリーム(半分のサイズ)に切り替わる必要のある単一イメージのコード・ストリームが他の半分で実行することである。これはしばしば不便である。図3(上)と図4(下)は、分離モードで、分離ストリームが他のサイドで全速で実行している間に一方のサイドで単一イメージのコード・ストリームを半速で実行する能力により特徴付けられる他の実施例を図示する。半速が意味することは、単一イメージのストリームのAサイド命令が例えば偶数サイクルで実行し、Bサイド命令が奇数サイクルに実行し、従って8命令の単一イメージ群を実行するのに少なくとも2サイクルかかることである。全てが並行化できない場合は、全8命令を実行するのにももちろん8サイクルかかる。同時に、全ての4命令が並行にディスパッチ可能であると想定すると、Bサイドは全速で新たな半分のサイズの(4命令)群を実行する。

【0034】本実施例は制御レジスタを2重化する必要がある、汎用レジスタの両組が一方のサイドに存在する必要がある(すなわち単一イメージのサイドは単一イメージAとBレジスタファイルの両方のコピーを必要とし、一方分離イメージBサイドはそれ自身のBサイド・レジスタファイルを必要とする)、従って一方のサイドの汎用レジスタファイルのサイズが2倍となる。また、交差レジスタファイル読取り、書込み及び述語の意味が変化し、単一イメージのサイドではその通常の機能を保持するが同一サイドの「他のサイクル」状態を指し、一方分離イメージ・サイドでは通常不正操作として処理される。単一イメージのサイドでは、「Aサイド」サイクル時に発生する交差ファイル操作は「Bサイド」を指し、逆もそうである。

【0035】単一イメージのサイドが他方のサイドからそれ自身の拡張レジスタファイル・イメージへ汎用レジスタファイル内容をコピーする必要があるという点で、本実施例は分離前にさらに作業を必要とする。又は別案として、別のレジスタファイル「C」を分離イメージ・

サイドに設け、適切な多重化と制御を与えて単一イメージ・サイドがA及びBファイルアクセス可能とし、一方分離サイドは通常のBファイルの代わりにCファイルにアクセスすることができる。このバージョンは分離前にBレジスタファイルを保存又はコピーする必要はない。

【0036】従来の場合と同様にこの場合には、どちらのメモリ（プログラム又はデータ）もサイズに関して変更されず、2つの命令ストリームは共存を可能とするようコード化されることが期待されていることに注意されたい。メモリ区域を拡大する又はこれを仮想的に拡大する（すなわち、キャッシュ）各種の方法が当該技術において公知であるが、望ましい実施例はプログラマがメモリ使用に絶えず注意を払い、必要に応じて分割することを必要とする（これはプログラマにとって無理な重荷としては考えられない）

【0037】図3に戻ると、本実施例はプログラムメモリ105への「2重」アクセスの必要性を処理するほかの方法を示す。この場合、アクセス優先度論理部303の制御下でメモリは各サイクルで全8命令パケット（256ビット）を出力する。この論理はその他の制御情報と共に両サイド110Aと110Bからフェッチ・プログラムカウンタ及び制御装置110Aと110Bを得て、各サイクルでどちらのサイドが命令パケット・フェッチを発行するかを選択する（もしあれば）。通常動作では、各サイドは1つおきのサイクル毎にその次の群を得る、すなわち偶サイクルにサイドAは8命令パケットを得て、奇サイクルにサイドBはその次のパケットを得る。どちらのサイドも1サイクルで4命令以上を消費できないため、FIFOバッファが各々装置307Aと307Bとして各サイドに設けられ、このバッファリングは各サイド・フェッチ・プログラムカウンタ及び制御論理110Aと110Bの制御下にある。この種のプリフェッチは当該技術において公知であり、第1実施例の2重ポート化の別案としてここに示す。

【0038】Bサイドを見ると、機能性は第1実施例（これは「分離」サイドで、Aサイドは「単一イメージ」サイドである）と基本的に同様で、箱322Bは、実行用の半パケットを用意し、これを「通常の」C6x（第1実施例のように）としてディスパッチする命令／デコード論理115内の回路を表示する。Bサイドに図示してある割込みマスク・ステア論理125は事実上分離されておらず、第1実施例のように動作する。

【0039】反対に、Aサイドはその機能を以下の様に拡張している。命令アダプタ217Aは、命令パケットをAサイド・パケット318AとBサイド・パケット318Bにソートする役割の命令ソート・アダプタ装置317により置換えられた。加えて、ある場合に各命令に追加ビットを追加することを含む命令のアダプタはここで実行される（すなわち、Bサイド・レジスタ・アドレス

0-15はAサイド・レジスタファイルの16-31であるレジスタ・アドレスB' と置換えられる）。元の順序と並行性を指示する追加ビットが各パケットとパケット中の個々の各命令に追加され、従ってパケットは「命令パケット」と指示される。制御論理319はAサイド又はB（B' と呼ばれる）サイド・サイクルがこのサイクルを実行すべきかを決定し、これをサイドAの残りに指示する。マルチプレクサ321は制御論理319の制御下でAサイド又はB' サイド・パケット（318A又は318B）のどちらかを選択する。選択したパケットはサイドBに示すように「通常の」C6xディスパッチ及びデコード論理により処理される。この流れにより、交互のAサイド・パケットとB' サイド・パケットがAサイド・パイプラインを流れ、通常速度の半分で行う全単一イメージ・パケットの効果を与える。Aサイドはサイクル当たり最大4演算のみを実行可能なため、全ての並行ビットをセットしていない任意の全パケットと同様に、単一サイドに4命令以上を有する全パケットは処理するために余分のサイクルがかかる。従って、余分な6サイクルまでが多分完全に直列なコードの場合に追加される。新たな各全パケットがAサイドで開始すると、パイプラインは時折シーケンス「 $\Psi A, B', A, A, B', \Psi$ 」を見、これは、第1パケットが1個以上のAサイド命令に接している又は全てのAサイド命令パケットを処理した時に発生し、発生する他のシーケンスは「 $\Psi A, B', B', A, B', \Psi$ 」で、これは、全てのB' サイド命令のパケットが1個以上のB' サイド命令とつながるパケットに続く時に発生する。もちろん上記シーケンスの両方が1回以上発生しうる、すなわち「 $\Psi A, B', B', B', \Psi, B', A, B', A, A, \Psi, A, B', \Psi$ 」のようなシーケンスを見ることもある。

【0040】以下の表はプロセッサ中の命令の粗の流れ順序を示し、「半速」に伴う性能差をさらに図示する。「通常のC6x流れ」という名称の列は一对の命令が標準（未分離）プロセッサを如何に流れるかを示し、「単一イメージAサイド」という名称の列は同様の命令が本発明の第2実施例の単一イメージ・サイドを如何に流れるかを示す。比較用に図示した「分離コードBサイド」という名称の最後の列は、半分のサイドの「全速」サイドの4半分（4）命令パケットの粗の流れを示す。列「通常のC6x流れ」と「単一イメージAサイド」の両方は命令群の以下の対を操作する。この例では {} 内の命令は単一サイクル（標準プロセッサで）内で並行に実行されるべきもので!!は前のサイクルのものと同様に並行に実行されるべき次ぎの命令が現れるべきであることを意味する（単一イメージ・サイドにのみ適用）

第1群： $A_{11} A_{12} \{B_{11} A_{13}\} B_{12} \{B_{13} B_{14}\} A_{14}$

第2群： $\{A_{21} B_{21} A_{22} B_{22}\} A_{23} \{A_{24} B_{23} B_{24}\}$

【表1】

| サイクル | 通常のC6x流れ | 単一イメージAサイド | 分離コードBサイド |
|------|--|---|--|
| 1 | A ₁₁ | A ₁₁ | {C ₁₁ C ₁₂ } |
| 2 | A ₁₂ | A ₁₂ | C ₁₃ |
| 3 | {B ₁₁ A ₁₃ } | B ₁₁ (can. 1 pair A. s & B. s) | C ₁₄ |
| 4 | B ₁₂ | A ₁₂ | {C ₂₁ C ₂₂ C ₂₃ C ₂₄ } |
| 5 | {B ₁₂ B ₁₃ } | B ₁₂ | C ₂₅ |
| 6 | A ₁₄ | {B ₁₂ B ₁₃ } | C ₂₆ |
| 7 | {A ₂₁ B ₂₁ A ₂₂ B ₂₂ } | A ₁₄ | C ₂₇ |
| 8 | A ₂₂ | {A ₂₁ A ₂₂ } | C ₂₈ |
| 9 | {A ₂₂ B ₂₂ B ₂₃ } | {B ₂₁ B ₂₂ } | C ₂₉ |
| 10 | | A ₂₃ | {C ₃₁ C ₃₂ } |
| 11 | | A ₂₄ | C ₃₃ |
| 12 | | {B ₂₃ B ₂₄ } | |

【0041】上記の表で、AとBサイド命令を共に実行する能力が第2実施例にないため、分離モードでは未分離モードより命令の組は実行に長くかかることが分かる。大量の更なる複雑性の追加によりこれは直せるが、半分の機能装置を有することによる制約は残る。また、この表は混合したAとBの並行群をシステムが如何に認識し、共に実行可能な全A及び全B群を作成するかを図示し、一方これは明らかな再順序付け(AはBの1実サイクル前に実行する)に関係し、プロセッサの設計は必要な場合に同一サイクルで実行しているかのよう2つのサイクル間で状態を保持する。

【0042】C6xで発生する唯一のストールはメモリ・バンクの矛盾によるものか、又はプログラム・メモリがキャッシュとして使用された場合キャッシュ・ミス時にもストールが発生する。望ましい実施例は、どちらかのサイドでストールが発生した時、通常のC6xのように全システム(両サイド)がストールする、これは制御を簡単にする。しかしながら、ストールに関する限り2サイドを完全に分離することも可能で、この場合一方の側のストールは他方に影響しない。

【0043】以後第3実施例と呼ばれる第2実施例のさらなる強化は、プロセッサ上で実行している通常の全サイズの全速コード・ストリームを全サイズの半速ストリームとしてAサイド上に「ロール」するための割込みハードウェアの能力を追加する。この「ロール」は全サイズ・ストリームが知ることなく発生し、その結果Bサイドを空けて割込み処理用に半サイズの全速コードを実行させる。これは以下のように実行される：

1. プロセッサ200が結合モード(未分離)で実行している時にマスクされない割込みを受取る。
2. 割込みマスク及びステア論理125により、プロセッサ200は上記実施例で記述したように分離モードへの進入を開始する。これはB及びB'レジスタファイルを使用することが望ましく、従って集団コピーは必要ない。必要な状態レジスタとその他の状態情報はAサイド130Aへコピーされる。Aサイド・フェッチ・プログラ

ムカウンタ及び制御装置110Aは不変のままで、切り替えが発生している間Aサイド130Aはストールする。これには数サイクルかかる。

3. Bサイド・フェッチ・プログラムカウンタ及び制御論理110Bには適切な割込みハンドラ・アドレスがロードされ開始される。割込み(又は他の短い「導入ルーチン」)が終了すると、これはプロセッサを再結合するポストアンブルを介してルーチンを出、Aサイドのスレッドは分離前と同様に全プロセッサを使用して続行する。

【0044】この能力のソフトウェア(すなわち、実時間オペレーティングシステム、RTOS)要件をここで考えてみると、時折Bサイド130Bをプリエンパットしたい割込みハンドラやその他の「時間依存」操作は、半サイド・コード語/装置で書かれる必要がある。これらの操作は、その他の小さなタスクを直後に実行するのでない限り、Aサイド130Aのプロセッサを再結合しなければならない。

【0045】RTOSのジョブはさらに複雑である、なぜならこれは2つの異なる型式のコード・ストリーム(全サイズ及び半サイズ)と3種のプロセッサ構成(未分離、分離した2つの半サイド、分離した1個の全サイドと1個の半サイド)をトレードオフしなければならない。コンパイラは、相互混合コード・ストリームを別々のサブストリームに分割し、しばらく分離して実行し、次いで必要に応じて以後再結合できることを理解し、使用する必要がある。コンパイラは、サイクル当たりのILPの量が閾値を越えた時に全サイズのコード・ストリームを発生しなければならない、それ意外では半サイズのコード・ストリームを発生しなければならない。

【0046】実時間オペレーティングシステムは、全サイズのストリームは限定された時間の間プロセッサの一方のサイドを放棄するよう決定し、従ってRTOSはこの期間の間未使用サイドで何かを実行可能であることを理解しなければならない。「ゲスト」タスクに利用可能な時間量を指示する方法がRTOSへ渡される(又は利用可能

な) 情報の一部として指示されなければならない。「ゲスト」スレッドがさらに時間を必要とし、十分重要的な場合、終了する間に全サイズのストリームを半速モードのAサイドへ「ロール」するよう決定でき、次いでサイドを再結合できることに注意されたい。

【0047】本明細書で記述した例に加えて、さらに他の例が当業者には認められる。従って、本実施例は詳細に記述したが、以下の特許請求の範囲に記載した発明の範囲から逸脱することなく各種の置換え、修正又は変更を上述した説明に加えうる。

【0048】以上の説明に関して更に以下の項を開示する。

(1) 超長命令語データプロセッサにおいて、複数データレジスタと、複数機能装置と、複数プログラムカウンタと、を含み、前記データプロセッサは、プログラムカウンタの所定のものの制御下で、データレジスタの選択されたものと機能装置の選択されたものによりプログラム命令が実行される第1モードと、第1群のデータレジスタと第1群の機能装置を使用して第1プログラムカウンタがプログラム命令の実行を制御し、前記第1群のデータレジスタから分離した第2群のデータレジスタと前記第1群の機能装置から分離した第2群の機能装置を使用して第2プログラムカウンタがプログラム命令の実行を制御する第2モードと、に従って選択的に動作可能な、超長命令語データプロセッサ。

(2) 第1項記載の超長命令語データプロセッサにおいて、複数データレジスタは複数データレジスタの半分のAサイドと複数データレジスタの半分の分離した組のBサイドから構成され、複数機能装置は複数機能装置の半分のAサイドと複数機能装置の半分の分離した組のBサイドから構成され、第1プログラムカウンタはAサイド・データレジスタとAサイド機能装置とに関係し、第2プログラムカウンタはBサイド・データレジスタとBサイド機能装置とに関係し、第2モードで、データプロセッサは2つの独立したプログラム命令ストリームを同時に処理可能な、超長命令語データプロセッサ。

(3) 第2項記載の超長命令語データプロセッサにおいて、複数制御レジスタであって、少なくとも1個の制御レジスタはAサイド部分とBサイド部分とを有する前記複数制御レジスタを含み、第1モードではデータレジスタと機能装置の動作は少なくとも1個の制御レジスタのAサイド部分により制御され、第2モードではAサイド・データレジスタとAサイド機能装置の動作は少なくとも1個の制御レジスタのAサイド部分により制御され、Bサイド・データレジスタとBサイド機能装置の動作は少なくとも1個の制御レジスタのBサイド部分により制御される、超長命令語データプロセッサ。

【0049】(4) 第2項記載の超長命令語データプロセッサにおいて、データプロセッサは、第2モードで、第1プログラムカウンタにより制御される複数データ

プロセッサの半分と機能装置の半分とを参照する命令のみを含む第1命令ストリームに回答し、Aサイド・データレジスタとAサイド機能装置を使用して実行され、第2モードで、第2プログラムカウンタにより制御される複数データプロセッサの半分と機能装置の半分とを参照する命令のみを含む第2命令ストリームに回答し、Bサイド・データレジスタとBサイド機能装置を使用して実行される、超長命令語データプロセッサ。

(5) 第2項記載の超長命令語データプロセッサにおいて、データプロセッサは、第2モードで、第1プログラムカウンタにより制御される任意の複数データプロセッサと任意の複数機能装置とを参照する命令を含む第1命令ストリームに回答し、(1) Aサイド・データレジスタとAサイド機能装置を参照する命令と(2) Bサイド・データレジスタとBサイド機能装置を参照する命令を交互にディスパッチすることによりAサイド機能装置を使用して実行され、第2モードで、第2プログラムカウンタにより制御される複数データプロセッサの半分と機能装置の半分とを参照する命令のみを含む第2命令ストリームに回答し、Bサイド機能装置を使用して実行される、超長命令語データプロセッサ。

(6) 第5項記載の超長命令語データプロセッサにおいて、さらにデータプロセッサの制御用の命令を記憶するプログラムメモリと、を含み、第1モードで、データプロセッサは第1プログラムカウンタに対応する各サイクルでプログラムメモリからNビットの命令をフェッチし、第2モードで、データプロセッサは交互のサイクルで、第1プログラムカウンタに対応するプログラムメモリからNビットの命令をフェッチし、第2プログラムカウンタに対応するプログラムメモリからNビットの命令をフェッチする、超長命令語データプロセッサ。

【0050】(7) 第5項記載の超長命令語データプロセッサにおいて、さらに、データプロセッサの制御用の命令を記憶するプログラムメモリと、を含み、第1モードで、データプロセッサは第1プログラムカウンタに対応する各サイクルでプログラムメモリからNビットの命令をフェッチし、第2モードで、データプロセッサは第1プログラムカウンタに対応するプログラムメモリからN/2ビットの命令をフェッチし、第2プログラムカウンタに対応するプログラムメモリからN/2ビットの命令をフェッチする、超長命令語データプロセッサ

(8) 第2項記載の超長命令語データプロセッサにおいて、第1モードと第2モードとの間及び第2モードと第1モードとの間の切替えは命令により制御される、超長命令語データプロセッサ。

(9) 第8項記載の超長命令語データプロセッサにおいて、第1制御レジスタ内の所定位置に第1組のビットを記憶してデータプロセッサは第1モードと第2モードとの間を切替え、第1制御レジスタ内の所定位置の第1組のビットからの第1組のビット差を記憶してデータプロ

セッサは第2モードと第1モードとの間を切替える、超長命令語データプロセッサ。

(10) 第8項記載の超長命令語データプロセッサにおいて、データプロセッサは分離命令の実行時に第1モードと第2モードとの間を切替え、データプロセッサは結合命令の実行時に第2モードと第1モードとの間を切替える、超長命令語データプロセッサ。

(11) 第2項記載の超長命令語データプロセッサにおいて、さらに、第1命令ストリーム又は第2命令ストリームのどちらが割り込みを受けるかを制御する命令を可能にする割り込みステア・マスク制御論理と、を含む、超長命令語データプロセッサ。

【0051】(12) 本発明は複数データレジスタ140A、140Bと、複数機能装置130A、130Bと、複数プログラムカウンタ110A、110Bとを含む超長命令語データプロセッサであり、第1又は第2モードのどちらかで選択的に動作可能である。第1モードでは、データプロセッサは単一命令ストリームを実行する。第2モードでは、データプロセッサは2つの独立したプログラム命令ストリームを同時に実行する。第2モードでは、データプロセッサはデータレジスタと機能装置の対応する半分にのみアクセスする2つの命令ストリームにตอบสนองしてもよい。または、データプロセッサは、(1) Aサイド・データレジスタとAサイド機能装置を参照する命令と(2) Bサイド・データレジスタとBサイド機能装置を参照する命令を交互にディスパッチすることにより、Aサイド機能装置を使用して全データプロセッサを参照する命令を含む第1命令ストリームにตอบสนองしてもよい。第1モードでは、データプロセッサは各サイクルでNビットの命令をフェッチする。第2モードでは、データプロセッサは交互のサイクルで交互のプログラムカウンタ用のNビットの命令をフェッチするか、又は第1及び第2プログラムカウンタの各々のN/2ビットをフェッチしてもよい。データプロセッサは第1命令ストリー

ム又は第2命令ストリームのどちらが割り込みを受けるかを制御するための命令を可能とする割り込みステア・マスク制御論理125を含む。

【図面の簡単な説明】

本発明の以上の及びその他の特徴は図面に図示される：

【図1】当該技術において既知のVLIWプロセッサの例である、テキサス・インストラメント社C62xファミリのDSPコアとプログラム及びデータメモリの簡略化したブロック線図である。

【図2】本発明の1実施例に従って変更された図1のプロセッサのブロック線図である。

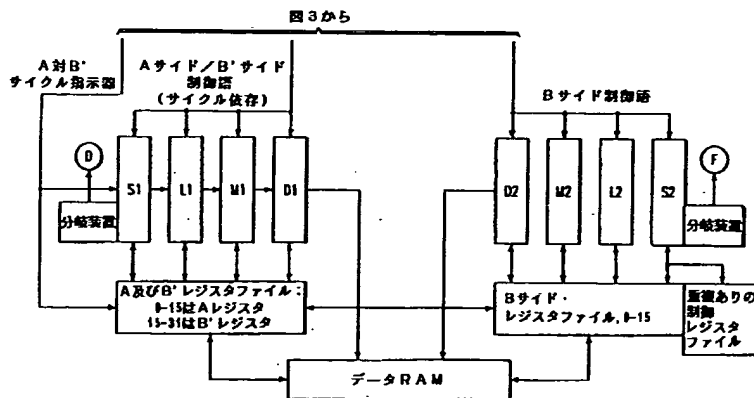
【図3】本発明の別な実施例により変更された図1のプロセッサの上半部である。

【図4】本発明の別な実施例により変更された図1のプロセッサの下半部である。

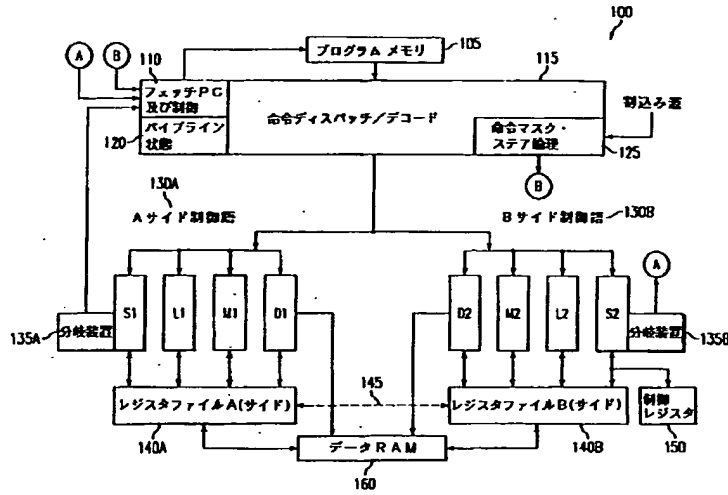
【符号の説明】

- 100 CPU
- 105 プログラムメモリ
- 160 データメモリ
- 130A Aサイド
- 130B Bサイド
- S、L、M、D 機能装置
- 140A、140B レジスタファイル
- 110、110A、110B フェッチ・プログラムカウンタ及び制御装置
- 115、115A、115B 命令ディスパッチ/デコード装置
- 135A、135B 分岐装置
- 125 割り込みマスク及びステア装置
- 120 バイプライン状態装置
- 145 レジスタ交差路
- 150、150A、150B 制御レジスタ
- 217A、217B 命令アダプタ

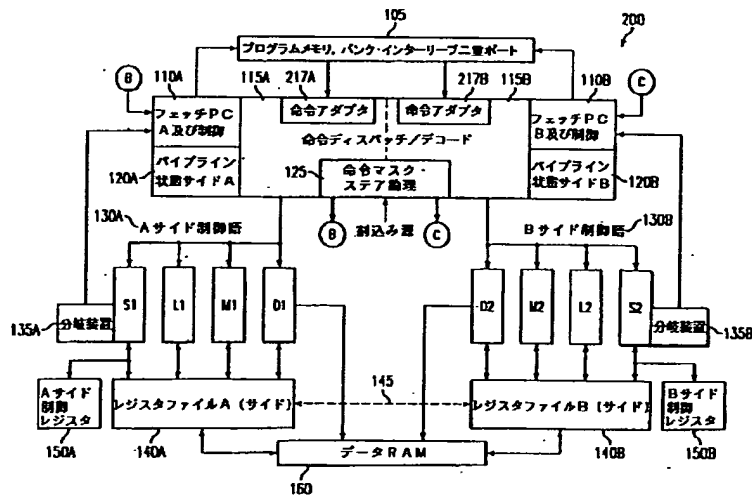
【図4】



【図1】



【図2】



【図3】

